

AMENDMENTS TO THE CLAIMS

Claims 1, 2, 15, 17-19, 32 and 34 are being amended. Claims 4-13, 21-30, 35-37 and 40-42 are being canceled. Claims 14, 31, 38 and 39 were previously canceled). All pending claims are reproduced below, including those that remain unchanged.

1. (Currently Amended) A method for extending an existing object oriented programming language, comprising the steps of:

selecting a program source file including a workflow definition created using an existing object oriented programming language, wherein the program source file includes a source code and classes therein and a workflow definition created using the existing object oriented programming language that is specified in the form of annotations to the source code and the classes; and,

extending [[an]] the existing object oriented programming language by adding at least one language construct defined by a second language so that the source code is extended with a plurality of workflow constructs defined by the second language, including adding an action construct representing an activity that allows a first software component written using the extended existing programming language to call an operation on a second software component written using the existing object oriented programming language; and

using a workflow program according to the workflow definition, including processing, using a computer including a processing device operating thereon, the action construct to allow the first software component written using the extended existing programming language to call the operation on the second software component written using the existing object oriented programming language.

2. (Currently Amended) A method according to claim 1, wherein:

the second language is a markup language ~~said existing programming language is an object oriented language.~~

3. (Original) A method according to claim 1, wherein:

said second language is XML.

4.-14. (Canceled)

15. (Currently Amended) A method according to claim 1, wherein:

~~said extending further comprises adding said language construct~~ is a multiple receive construct that allows a software component written using the extended existing programming language to wait on multiple input events received.

16. (Original) A method according to claim 15, wherein:

said multiple receive construct further allows said software component proceed on a particular branch of program execution, based on the input event that occurred first within the said multiple input events.

17. (Currently Amended) A method according to claim 1, wherein:

~~said extending further comprises adding said language construct~~ is a looping construct with ordering of messages received, representing looping functionality, wherein the ordering allows said messages to be received in an order.

18. (Currently Amended) A computer system capable of extending an existing programming language, comprising:

a computer including a processing device operating thereon;

a program source file stored on the computer, wherein the program source file includes a workflow definition created using an existing object oriented programming language stored in the computer system, wherein the program source file includes a source code and classes therein and a workflow definition created using the existing object oriented programming language that is specified in the form of annotations to the source code and the classes, and wherein the source code is extended with a plurality of workflow constructs defined by a second language, including an action construct representing an activity that allows a first software component written using the extended existing programming language to call an operation on a second software component written using the existing object oriented programming language; and,

means for extending an existing programming language by adding at least one language construct defined by a second language using a workflow program according to the workflow

definition, including processing, using a computer including a processing device operating thereon, the action construct to allow the first software component written using the extended existing programming language to call the operation on the second software component written using the existing object oriented programming language.

19. (Currently Amended) A computer system according to claim 18, wherein:
the second language is a markup language ~~said existing programming language is an object oriented programming language.~~

20. (Previously presented) A computer system according to claim 18, wherein:
said second language is XML.

21.-31. (Canceled)

32. (Currently Amended) A computer system according to claim 18, wherein:
the source code is also extended with ~~said language construct~~ is a multiple receive construct that allows a software component written using the extended existing programming language to wait on multiple input events received.

33. (Previously presented) A computer system according to claim 32, wherein:
said multiple receive construct further allows said software component proceed on a particular branch of program execution, based on the input event that occurred first within the said multiple input events.

34. (Currently Amended) A computer system according to claim 18, wherein:
the source code is also extended with ~~said language construct~~ is a looping construct with ordering of messages received, representing looping functionality, wherein the ordering allows said messages to be received in an order.

35.-42. (Canceled)